Mineração de Dados em Biologia Molecular

Pré-processamento de dados

André C. P. L. F. de Carvalho
Monitor: Valéria Carvalho

Tópicos

- Introdução
- Amostragem
- Qualidade de Dados
  - Limpeza de Dados
- Transformação de dados
- Seleção de atributos

Pré-processamento

- Prepara os dados para uso de algoritmo de modelagem
- Procura melhorar desempenho do algoritmo
  - Custo
    - Tempo
    - Memória
  - Qualidade

Exemplo

- Primeiro passo:
  - Eliminar atributos irrelevantes

Nome	Febre	Enjôo	Mancha	Dor	Salário	Diagnóstico
João	sim	sim	pequena	sim	1000	doente
Pedro	não	não	pequena	não	1100	saudável
Maria	sim	sim	grande	não	600	saudável
José	sim	não	pequena	sim	2000	doente
Ana	sim	não	grande	sim	1800	saudável
Leila	não	não	grande	sim	900	doente

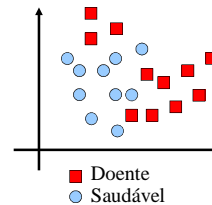
Amostragem de dados

- Seleção de objetos
- Base de dados grande
  - Algoritmo de AM não precisa usar todo conjunto de dados
  - Eficiência X acurácia
- Amostra
  - Pode levar à mesma acurácia com um esforço computacional menor
  - Deve ser representativa

Amostragem de dados

- Amostra representativa
  - Aproximadamente as mesmas propriedade de interesse do conjunto de dados original
    - Ex.: $\text{média}_{\text{pop-original}} = \text{média}_{\text{amostra}}$
  - Deve fornecer uma estimativa da informação contida na população original
  - Uso da amostra deve ter efeito semelhante ao uso de toda a população
    - Não é possível garantir que isso ocorra
      - Técnicas de amostragem aumentam as chances

Amostragem de dados

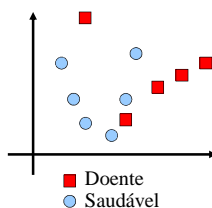
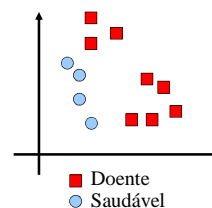
- Tipos de amostragem
  - Amostragem aleatória simples
  - Amostragem estratificada
  - Amostragem progressiva

Amostragem de dados

Doente
Saudável

Amostragem de dados

Doente
Saudável

Amostragem de dados

Doente
Saudável

Amostragem de dados

- Amostragem progressiva
  - Começa com pequenas amostras
    - Progressivamente aumenta tamanho da amostra
      - Enquanto acurácia do modelo preditivo aumentar
    - Confirmar com outras amostras de tamanho semelhante à escolhida
    - Boa estimativa de um bom tamanho

Dados desbalanceados

- Quando número de exemplos varia para as diferentes classes
  - Natural em alguns domínios
  - Problema com geração / coleta de dados
- Várias técnicas de AM não conseguem lidar com esse problema
  - Tendência a classificar na(s) classe(s) majoritária(s)
- Alternativa: balanceamento artificial

Qualidade de dados

- Em geral, dados não foram gerados para uso em AM
  - Produzidos para outros propósitos
  - Frequentemente apresentam problemas
- Algoritmos de AM precisam geralmente de dados "limpos"
  - Entra lixo, sai lixo
  - Problemas nos dados precisam ser detectados e corrigidos
    - Limpeza de dados

Qualidade de dados

- Problemas podem ocorrer nos processos de medições e na coleta de dados
- Erros podem ter causa
  - Sistemática
    - Mais fácil de detectar e corrigir
  - "Aleatória"

Qualidade de dados

- Exemplos de causas:
  - Falha humana
  - Falha no processo de coleta de dados
  - Limitações do dispositivo de medição
  - Má fé
  - Valor do atributo alvo muda com o tempo

Qualidade de dados

- Consequências:
  - Valores ou objetos inteiros podem ser perdidos
  - Objetos espúrios ou duplicados podem ser obtidos
    - Ex.: diferentes registros para mesma pessoa que morou em endereços diferentes
  - Inconsistências
    - Ex.: pessoa com 2 m pesando 10 Kg, idade e data de nascimento

Limpeza

- Correção de erros detectados nos dados
- Deve lidar com:
  - Dados com ruídos
  - *Outliers*
  - Dados incompletos ou atributos com valores ausentes
  - Dados inconsistentes
  - Dados redundantes

Ruídos

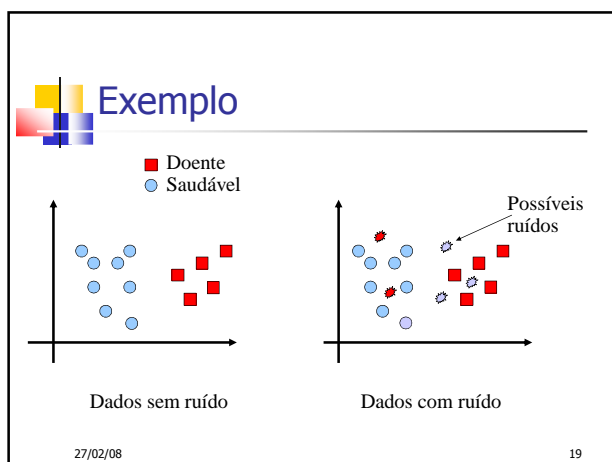
- Dados com ruídos podem levar a um super-ajuste do modelo
- Não é possível ter certeza de que um valor apresenta ruído
  - Tem-se apenas um indício, a menos que seja inconsistente
  - *Outliers* podem sugerir a presença de ruído
- Nos atributos de entrada ou no atributo alvo
  - Consequências diferentes

Exemplo

Doente
Saudável

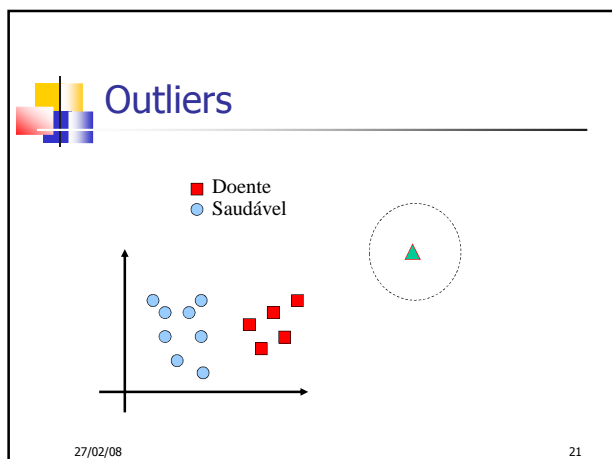
Possíveis ruídos

Dados sem ruído

Dados com ruído

Outliers

- Existem várias definições
- Objetos ou valores anômalos
  - Objetos que têm características diferentes da maioria dos demais objetos
    - Valores de um atributo que destoam dos valores típicos para o atributo
- Ao contrário de ruídos, *outliers* podem ser objetos ou valores legítimos
  - Em várias aplicações, objetivo é encontrar *outliers*

Outliers

Doente
Saudável

Valores ausentes

- Não é raro um objeto não ter o valor de um ou mais atributos
- Possíveis causas:
  - Atributo não foi considerado quando os primeiros dados foram coletados
  - Desconhecimento do valor do atributo por ocasião do preenchimento
  - Distração, mal entendido ou declinamento na hora do preenchimento
  - Não necessidade ou obrigação de apresentar um valor para atributo(s) de algumas instâncias
  - Inexistência de valor para o atributo em algumas instâncias
  - Problema com dispositivo / processo de coleta

Valores ausentes

- Alternativas
  - Ignorar valores ausentes
    - Utilizar apenas os que estão presentes nas instâncias consideradas
      - Ex.: Menos atributos na medida de distância
    - Modificar algoritmo para lidar com valores ausentes
  - Descartar exemplos com atributos que apresentem valores ausentes
  - Estimar valores ausentes

Valores ausentes

- Descartar exemplos
  - Geralmente empregada quando um dos atributos ausentes é o atributo classe
  - Não é indicada quando:
    - Ocorre com poucos atributos do exemplo
    - Número de atributos com valores ausentes varia muito entre os exemplos com esse problema
    - Há risco de descartar dados importantes

Valores ausentes

- Estimativa de valores ausentes
  - Utilizar algum método ou heurística para automaticamente definir valores
    - Alternativa mais utilizada
    - Diferentes abordagens podem ser seguidas

Valores ausentes

- Heurísticas para estimativa:
  - Criação de um novo valor
    - Dados categóricos nominais (sem ordem)
  - Média (mediana, moda) de todos os valores do atributo
    - Para série de valores, entre valores anterior e posterior
    - Moda = valor ou intervalo mais freqüente
  - Média (mediana, moda) dos vizinhos mais próximos
  - Valor induzido por algum estimador
    - Valor presente em exemplos semelhantes

Valores ausentes

- Observações
  - Em alguns casos, a ausência de valor é uma informação importante sobre a instância
  - Existem situações em que o valor precisa estar ausente
    - Ex.: Resultado de exame pré-natal para paciente do sexo masculino
      - Atributo número de partos para paciente do sexo masculino pode ter valor 0
    - Ao invés de ausente, é um valor inexistente
- Difícil tratar de forma automática

Exercício

- Tratar dos valores ausentes da tabela abaixo

Nome	Profissão	Nível	Peso	Altura	Salário	Situação
João	Encanador	Médio	70	180	3000	adimplente
Lia		Superior	200	174	7000	inadimplente
Maria	Advogado	Médio		180	600	adimplente
José	Médico	Superior	100		2000	inadimplente
Sérgio	Bancário		82	178	5000	inadimplente
Ana	Professor	Fundam.	77	188	1800	adimplente
Luísa	Médico	Superior	100	36	2000	inadimplente
José	Médico	Médio	340		800	

Valores inconsistentes

- Dados podem conter valores inconsistentes
  - Atributos preditivos
    - Ex. Dados com código postal invalido para o nome de rua especificado
      - Erro / engano
      - Proposital (fraude)
  - Atributo alvo
    - Podem levar a exemplos conflitantes
      - Ex.: valores iguais para atributos de entrada e diferentes para atributo de saída

Valores inconsistentes

- Algumas inconsistências são de fácil detecção
  - Violação de relações conhecidas entre atributos
    - Ex.: Valor de atributo A é sempre menor que valor de atributo B
  - Valor inválido para o atributo
    - Ex.: altura com valor negativo
  - Em outros casos, informações adicionais precisam ser verificadas

Valores redundantes

- Valores que não trazem informação nova
- Dados (quase) duplicados
  - Ex.: Pessoas em diferentes BDs com mesmo endereço e pequenas diferenças nos nomes
- Deduplicação
  - Detectar e eliminar (ou combinar) duplicações
  - Cuidado para não eliminar ou combinar dados que representam objetos diferentes

Exemplo

- Dados redundantes

Nome	Febre	Enjôo	Mancha	Dor	Salário	Diagnóstico
João	sim	sim	pequena	sim	1000	doente
Pedro	não	não	pequena	não	1100	saudável
Maria	sim	sim	grande	não	600	saudável
José	sim	não	pequena	sim	2000	doente
Sérgio	não	não	pequena	não	1100	saudável
Ana	sim	não	grande	sim	1800	saudável
Leila	não	não	grande	sim	900	doente
Marta	sim	não	pequena	sim	2000	doente

Exemplo

- Dados redundantes

Nome	Febre	Enjôo	Mancha	Dor	Salário	Diagnóstico
João	sim	sim	pequena	sim	1000	doente
Pedro	não	não	pequena	não	1100	saudável
Maria	sim	sim	grande	não	600	saudável
José	sim	não	pequena	sim	2000	doente
Sérgio	não	não	pequena	não	1100	saudável
Ana	sim	não	grande	sim	1800	saudável
Leila	não	não	grande	sim	900	doente
Marta	sim	não	pequena	sim	2000	doente

Exercício

- Definir problemas existentes na tabela abaixo:

Nome	Profissão	Nível	Peso	Altura	Salário	Situação
João	Encanador		70	180	3000	adimplente
Lia	Médico	Superior	200	174	7000	inadimplente
Maria	Advogado	Médio	90	180	600	adimplente
José	Médico	Superior	200	174	7000	inadimplente
Sérgio	Bancário	Superior	82	178	5000	inadimplente
Ana	Professor	Fundam.	77	188	1800	adimplente
Luísa	Médico	Superior	100	-6	2000	inadimplente

Transformação de dados

- Conversão de valores numéricos para simbólicos
- Conversão de valores simbólicos para numéricos
- Binarização
- Normalização de valores numéricos
- Tradução de atributos

Conversão de valores simbólicos

- Algumas técnicas trabalham apenas com valores numéricos
  - Valores simbólicos precisam ser convertidos para numéricos
- Conversão depende de:
  - Ordenação dos valores
    - Presente ou ausente
  - Número de valores
    - = 2 (binários) ou > 2

Conversão ordinal para binário

- Codificar para valor inteiro positivo
  - Ex. Pequeno (1), médio (2) e grande (3)
- Algumas técnicas trabalham apenas com valores binários
  - Codificar cada valor por um vetor binário
    - Código cinza:
      - 000, 010, 011, 001, 101, 111, 110, 100
    - Código termômetro:
      - 001, 011, 111

Conversão nominal para binário

- Codificações
  - 1-de-n
    - Codificação canônica
    - Moda = posição com maior número de valores 1
    - Valores escalares podem virar vetores longos
  - m-de-n
    - Dos n valores, m são iguais a 1
    - Escolha de código

Bioinformática

- Análise de sequências de nucleotídeos ou de aminoácidos
  - Grande número de atributos
    - Valor definido de um alfabeto de 4 (nucleotídeos) ou de 20 (aminoácidos) possíveis valores
    - Ordem dos valores dos atributos na seqüência é importante
      - Mas valores não são ordenados
  - Reconhecimento de genes, previsão de estrutura de proteínas

Bioinformática

- Alternativas para codificação numérica de seqüências de bioinformática
  - Código 1-de-n para cada valor
  - Freqüência com que cada valor aparece
  - Freqüência com que subsequência de n valores aparece
  - Dividir a seqüência em m trechos e utilizar a freqüência dos valores em cada trecho
    - Preserva parte da ordem

Conversão numérico para ordinal

- Discretização de valores
  - Transformar valores numéricos em intervalos ou categorias
- Sub-tarefas
  - Definição do número de categorias
    - Geralmente feito pelo usuário
  - Definição de como mapear valores dos atributos contínuos para essas categorias
    - Definição do frequência/largura dos intervalos
    - Geralmente feito por um algoritmo

Pseudo códigos

- Imagine que um atributo seja nome de país
  - Existem 193 países (192 representados na ONU + Vaticano)
  - Alternativa de codificação:
    - Transformar valores nominais em valores numéricos utilizando a codificação 1-de-n

Exemplo

- Atributo = nome de país
  - 193 (192 representados na ONU + Vaticano)
  - Maldição da dimensionalidade
  - Grande parte dos elementos possui valor 0
    - Esparsos

1	0	0	•••	0
1	2			193

0	1	0	•••	0
1	2			193

⋮

0	0	0	•••	1
1	2			193

Exemplo

- Outra alternativa:
  - Transformar 193 atributos em 4 (10) pseudo-atributos
    - Continente: 7 valores
    - PIB: 1 valor
    - População: 1 valor
    - Área: 1 valor

Transformação de atributos

- Valor numérico de um atributo pode precisar ser transformado em outro
  - Limites de valores para atributos distintos podem ser muito diferentes
    - Evitar que um atributo predomine sobre outro
      - A menos que isso seja importante
  - Grande intervalo de variação de valores
    - Pode aumentar custo computacional

Transformação de atributos

- Aplicada aos valores de um dado atributo de todos os objetos
  - Ex.: supor que apenas a magnitude do valor de um atributo é importante
    - Converter valor de todos os atributos é para o valor absoluto
    - -4, 5 e –2 se tornam 4, 5 e 2
  - Variações
    - Funções simples
    - Normalização

Funções simples

- Uma função matemática simples é aplicada a cada valor do atributo
  - Possíveis transformações:
    - X^k, log x, e^x, $\sqrt{x}$, 1/x, seno(x) e |x|
      - Funções sqrt, log e 1/x aproximam uma distribuição Gaussiana
  - Função $\log_{10}$ é usada para comprimir dados com um grande intervalo de valores

Normalização

- Faz com que conjunto de valores de um atributo tenha uma dada propriedade
- Alternativas
  - Pela amplitude
    - Re-escalar
    - Padronizar
  - Pela distribuição

Re-escala

- Para re-escalar os valores de um atributo:
  1. Adicionar ou subtrair uma constante
  2. Multiplicar ou dividir por uma constante
- Utilizado para mudar intervalo de valores dos dados
  - Permite converter todos os valores de um atributo para o intervalo [0, 1]

$$d' = \frac{(d - \min_d)}{(\max_d - \min_d)}$$

Exercício

- Re-escalar os valores 12, 5, 4, 10, 20, 3 para o intervalo [-1, +1]

Padronização

- Para padronizar os valores de um atributo:
  1. Adicionar ou subtrair uma medida de localização
  2. Multiplicar ou dividir por uma medida de escala
- Se os valores têm uma distribuição Gaussiana
  - Subtrair a média
  - Dividir pelo desvio padrão
  - Produz conjunto de valores com distribuição normal (0,1)

Exercício

- Converter os seguintes valores numéricos utilizando re-escala e padronização

Valores	Re-escala	Padronização
3		
9		
5		
11		
5		
7		

$$\text{var}(v) = \frac{1}{n-1}\sum_{i=1}^{n}(v_i - \overline{v})^2$$

Exercício

Não existe relação de ordem para os tipos de dor

- Converter os dados abaixo para valores numéricos no intervalo [0, 1]

Febre	Enjôo	Mancha	Dor	Diagnóstico
baixa	sim	pequena	A	doente
média	não	média	C	saudável
alta	sim	grande	B	saudável
alta	não	pequena	A	doente
baixa	não	grande	D	saudável
média	não	sem	C	doente

Conversão de valores numéricos

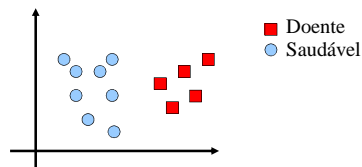
- É preferível padronizar a re-escalar
- Atributos mais importantes podem ter limites maiores
  - Padronizar
  - Re-escalar
- Normalização pela distribuição
  - Muda escala de valores
    - Ex.: função *log*

Tradução

- Ocorre devido a limitações no formato utilizado para armazenar o atributo
  - Alguns algoritmos podem ter dificuldades com formato original
  - Exemplos
    - Conversão de hora para valor inteiro
    - Conversão de data para valor inteiro
    - Conversão de rua para código postal

Maldição da dimensionalidade

- Supor que dados são representados por pontos em um hipervolume
  - Valores dos atributos são os valores das coordenadas

Doente
Saudável

Maldição da dimensionalidade

- Hipervolume cresce exponencialmente com a adição de novos atributos
  - Instâncias formadas por 1 atributo com 10 possíveis valores: 10 possíveis objetos
  - Instâncias formadas por 5 atributos com 10 possíveis valores: 10^5 possíveis objetos
  - Problemas com poucos exemplos e muitos atributos:
    - Dados se tornam muito esparsos

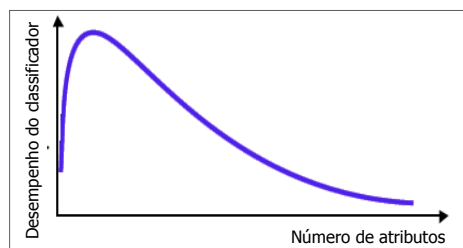
Maldição da dimensionalidade

- Dados esparsos
  - Sem exemplos em várias das regiões do espaço de objetos
  - Instâncias parecem equidistante
    - Prejudica o desempenho de algoritmos que medem similaridade de dados por distância

Maldição da dimensionalidade

- Número de exemplos necessários para manter desempenho
  - Cresce exponencialmente com o número de atributos
- Na prática, o número de exemplos de treinamento é fixo
- Redução de dimensionalidade

Maldição da dimensionalidade

Desempenho do classificador
Número de atributos

Redução de dimensionalidade

- Alguns conjuntos podem ter um número muito grande de atributos
  - Ex.: objeto é um vetor com frequência de cada palavra que aparece em um texto
- Reduzir dimensão
  - Agregação de atributos
    - Criar novos atributos que são uma combinação dos atributos originais
  - Seleção de atributos

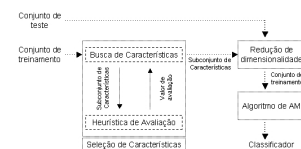
Seleção de atributos

- Permite
  - Identificar atributos importantes
  - Melhorar desempenho de algoritmo de para indução de modelos
  - Minimizar os efeitos de ruídos
  - Reduzir custo de coleta de dados
    - Aumentar acesso à tecnologia

Seleção de atributos

- Abordagens
  - Embutida
    - Seleção é feita pelo algoritmo de AM
  - Filtro
  - *Wrapper*
- Heurísticas
  - Ordenação
  - Subconjunto

Filtros

- Seleção de atributos independe do algoritmo de AM utilizado
  - Ex.: verifica co-relação entre atributos

Filtros

- Vantagens
  - Não depende do algoritmo de AM
    - Os atributos selecionados podem ser utilizados por diferentes algoritmos de AM
  - Baixo custo computacional
    - Podem ser muito rápidos
  - Conseguem lidar de forma eficiente com uma grande quantidade de dados

Filtros

- Desvantagens
  - Ignora interação com o algoritmo
    - Não levar o viés do algoritmo em consideração pode levar a modelos pouco eficientes
  - Pode ignorar dependências entre atributos

Wrappers

- Utilizam o algoritmo de AM para selecionar atributos
  - Ex. Atributos que levaram a menos erros de classificação para uma rede MLP

Conjunto de teste
Conjunto de treinamento
Busca de Características
Subconjunto de Características
Valor de avaliação
Avaliação de Características
Subconjunto de Características
Classificador
Algoritmo de AM
Seleção de Características
Subconjunto de Características
Redução de dimensionalidade
Conjunto de treinamento
Algoritmo de AM
Classificador

Wrappers

- Vantagens
  - Melhor conjunto para um dado algoritmo
    - Pode selecionar também melhor número de atributos
  - Geralmente melhora desempenho obtido pelo algoritmo

Wrappers

- Desvantagens:
  - Risco de *overfitting*
  - Desempenho depende do algoritmo de indução
  - Custo computacional elevado, por causa do grande número de execuções do algoritmo
    - Nem sempre, existem estratégias eficientes
  - Precisa ser repetido quando um novo algoritmo de AM for utilizado

Ordenação X Seleção

Ordenação

Atributos originais

1	2	3	4	5	6	7	8	9	10
1	2	3	4	5	6	7	8	9	10

Atributos ordenados

4	7	2	6	9	1	10	5	8	3
1	2	3	4	5	6	7	8	9	10

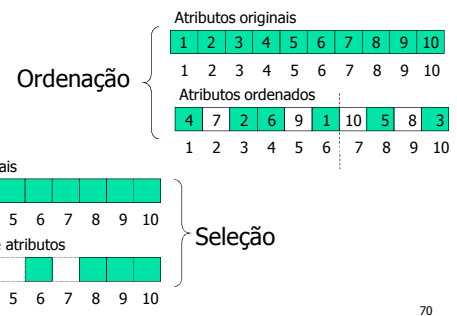
Seleção

Atributos originais

1 2 3 4 5 6 7 8 9 10

Subconjunto de atributos

1 2 3 4 5 6 7 8 9 10

27/02/08

Exercício

- Ordenar os atributos mais importantes para o diagnóstico de pacientes

Febre	Enjôo	Mancha	Dor	Diagnóstico
1	1	1	1	0
0	1	0	1	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	0	0

Exercício

Febre	Enjôo	Manchas	Dores	Diagnóstico
1	0	0	1	0
0	1	0	1	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	0	0

Enjôo: 5/6
Manchas: 4/6
Febre: 3/6
Dores: 3/6

Ranking:
1- Enjôo
2- Manchas
3- Febre
4- Dores

Exercício

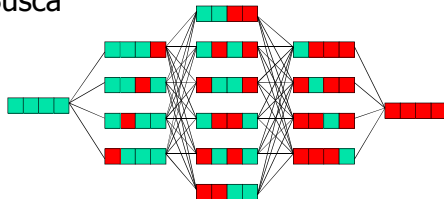
- Selecionar o subconjunto de atributos mais importantes para o diagnóstico de pacientes
  - *Wrapper*

Febre	Enjôo	Manchas	Dores	Diagnóstico
1	1	0	1	0
0	1	0	0	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	1	0

Exercício

Febre	Enjôo	Mancha	Dor	Diagnóstico
1	1	0	1	0
0	1	0	0	1
1	1	1	0	1
1	0	0	1	0
1	0	1	1	1
0	0	1	1	0

Febre ⊗ mancha
└ coincide

Seleção de subconjunto

Busca

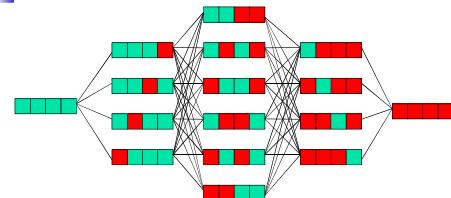
Espaço de busca com quatro atributos (dimensões)

Seleção de subconjunto

- Quatro aspectos precisam ser tratados:
  - Ponto de início da busca e da geração de subconjuntos
  - Estratégia de busca
  - Estratégia de avaliação
  - Critério de parada

Geração de subconjuntos

- Existem quatro alternativas
  - Geração para trás (*backward generation*)
    - Começa com todos os atributos e remove um por vez
  - Geração para frente (*forward generation*)
    - Começa sem nenhum atributo e inclui um atributo por vez
  - Geração bidirecional (*bidirectional generation*)
    - Busca pode começar em qualquer ponto e atributos podem ser adicionados e removidos
  - Geração estocástica (*random generation*)
    - Ponto de partida da busca e atributos a serem removidos ou adicionados são decididos de forma estocástica

Geração de subconjuntos

Backward
Feedforward
Bidirecional

Estratégia de busca

- Define o algoritmo usado para realizar a busca
  - Busca completa (exponencial ou exaustiva)
    - Avalia todos os possíveis subconjuntos
  - Busca heurística (sequencial)
    - Utiliza regras e métodos para conduzir a busca
    - Não garante que uma solução ótima seja encontrada
  - Busca não-determinística
    - Relacionado com a geração estocástica
    - Boa solução pode ser encontrada antes do final da busca
    - Não garante ótimo

Considerações finais

- Pré-processamento
- Amostragem
- Limpeza de dados
- Transformação de dados
- Redução do número de atributos

Perguntas

Exercício

- Escolher 3 conjuntos de dados da UCI e, para cada conjunto
  - Aplicar uma técnica de amostragem dos dados
  - Aplicar técnicas para limpeza de dados
  - Criar uma variação com todos os atributos numéricos
  - Criar uma variação com todos os atributos simbólicos
  - Selecionar atributos usando uma técnica baseada em filtro e uma baseada em wrapper